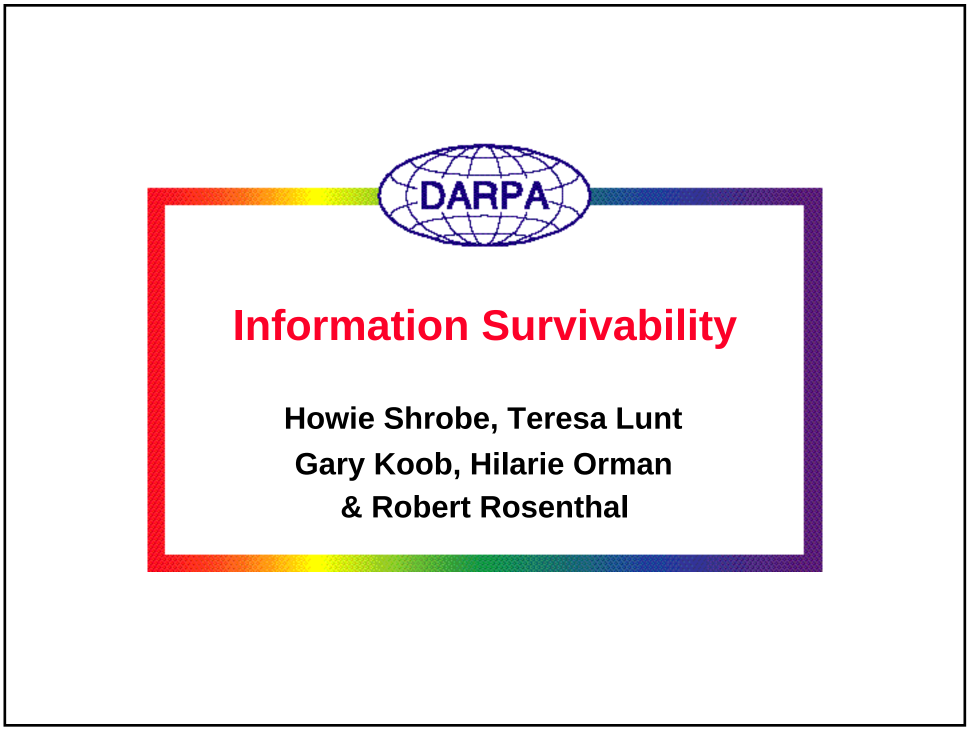
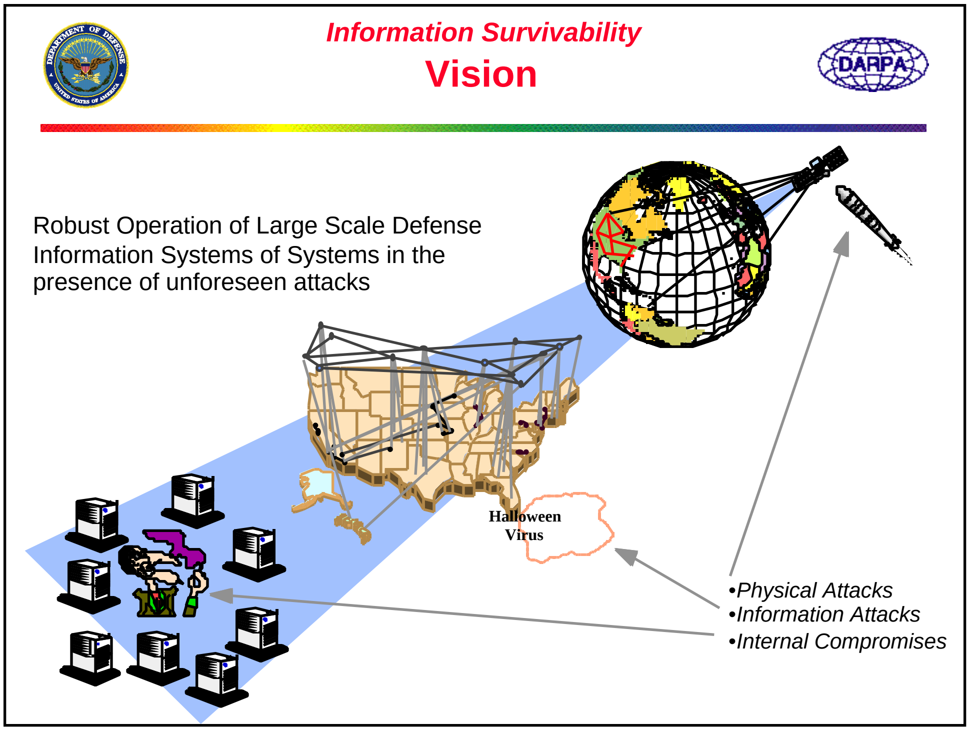


Information Survivability

Howie Shrobe, Teresa Lunt
Gary Koob, Hilarie Orman
& Robert Rosenthal



Information Survivability

Vision

Robust Operation of Large Scale Defense Information Systems of Systems in the presence of unforeseen attacks

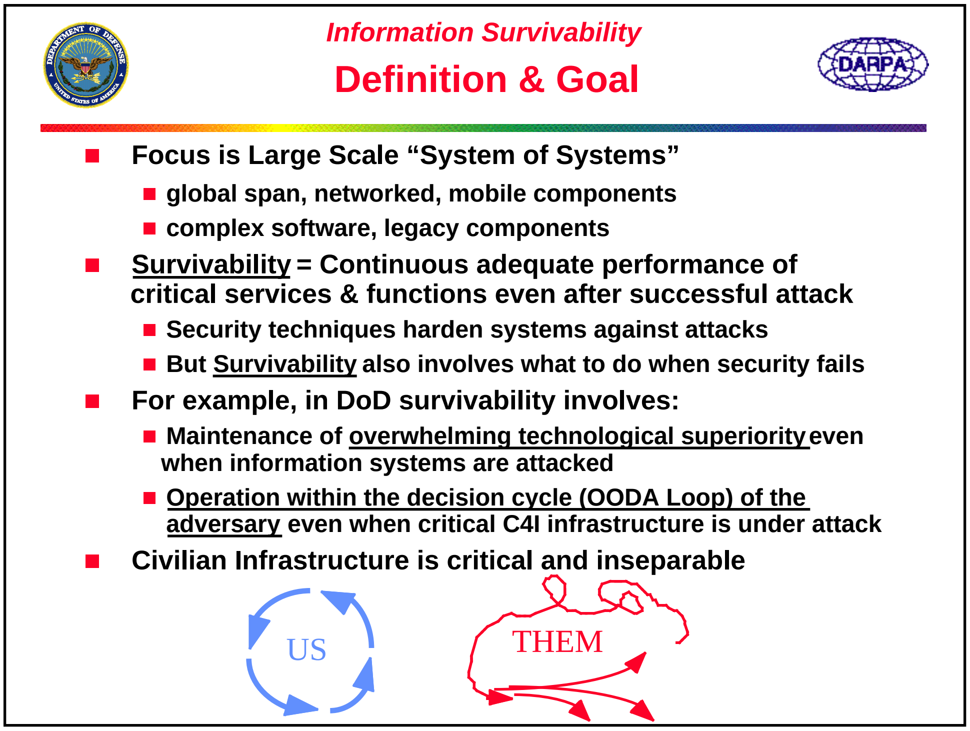
- *Physical Attacks*
- *Information Attacks*
- *Internal Compromises*

DoD's strategy of overwhelming technological superiority has led it to a reliance on highly integrated & complex military information systems. These systems interoperate with and rely on components from the commercial communications and computing infrastructure. In addition, DoD relies on the commercial transportation, power and aircraft control systems to achieve many aspects of its mission with the CONUS.

All of these systems are vulnerable to one degree or another: Their physical components can be attacked; they can be penetrated by unauthorized outsiders and they can be subverted by compromised malicious internal users with high degrees of authorization.

The global scale of the internet means that attacks on information systems within CONUS can be conducted from virtually anywhere in the world.

It is the goal of this research initiative to develop technology that will guarantee that these critical information systems continue to function adequately in the face of any of these forms of attack, even when the precise type of attack has not been anticipated.



Information Survivability

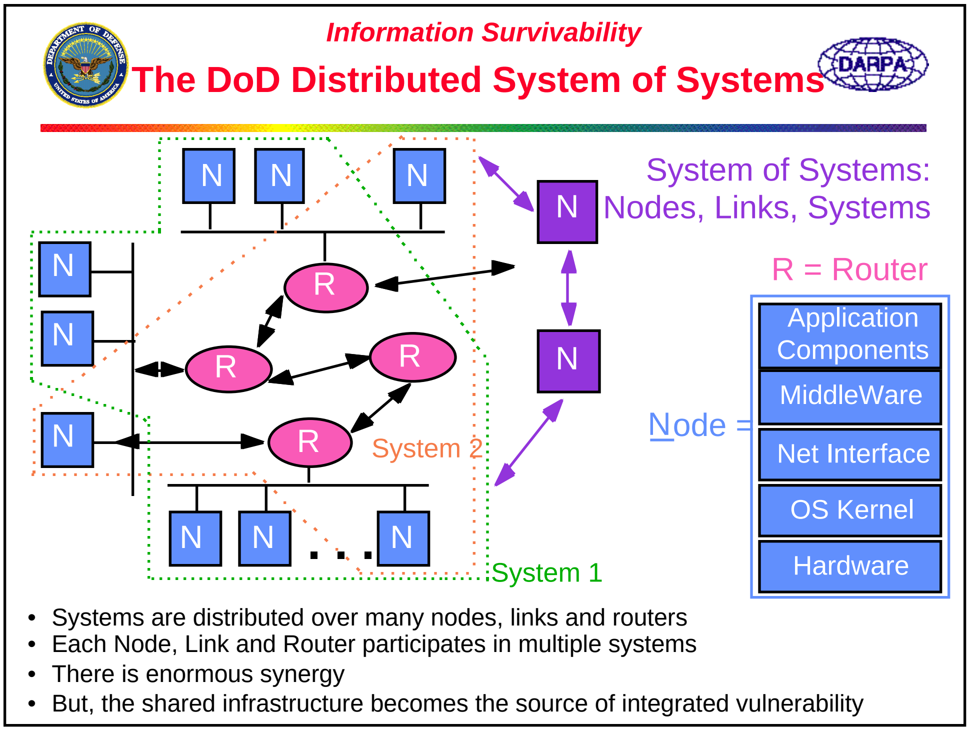
Definition & Goal

- Focus is Large Scale "System of Systems"
  - global span, networked, mobile components
  - complex software, legacy components
- Survivability = Continuous adequate performance of critical services & functions even after successful attack
  - Security techniques harden systems against attacks
  - But Survivability also involves what to do when security fails
- For example, in DoD survivability involves:
  - Maintenance of overwhelming technological superiority even when information systems are attacked
  - Operation within the decision cycle (OODA Loop) of the adversary even when critical C4I infrastructure is under attack
- Civilian Infrastructure is critical and inseparable

The focus of this research effort is "Survivability" of DoD "systems of systems"; these are large scale, complex distributed systems with global span. The systems involved are very long lived; they rely on legacy components, many of which were not designed with survivability as a prime consideration. However, it isn't a viable option to replace these components and do the whole thing over again. The cost would be prohibitive and by the time we were done, the world will probably have changed enough that the components will still be found wanting.

So the approach cannot be to "build it right the first time", components will have design flaws and security holes. This isn't to say that we should be cavalier about the use of security techniques; whenever possible we should harden our systems to the greatest degree affordable. Instead our goal should be "Survivability", the ability to continue performing critical tasks at an adequate level even when there are successful attacks launched. It is our goal to develop design techniques that will guarantee that our systems are survivable in this sense.

For DoD, this means at the most general level that we can maintain our overwhelming technological superiority even when an adversary intentionally attacks our information systems. In particular, it means that we need to guarantee that our Command and Control systems continue to allow to function the decision cycle of our enemies.



Information Survivability

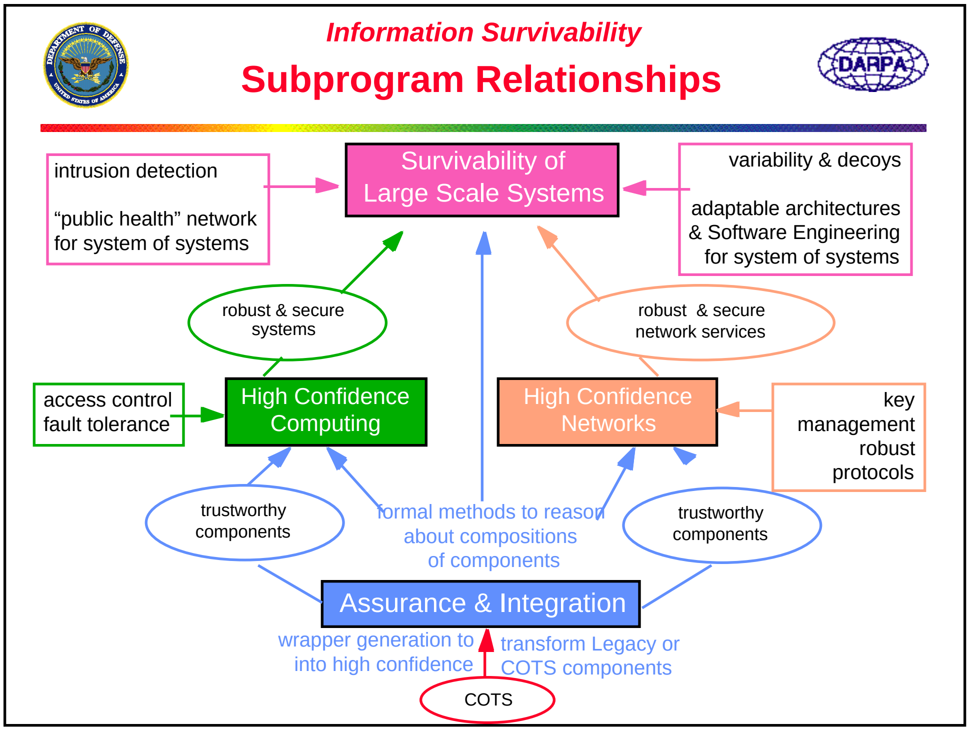
The DoD Distributed System of Systems

- Systems are distributed over many nodes, links and routers
- Each Node, Link and Router participates in multiple systems
- There is enormous synergy
- But, the shared infrastructure becomes the source of integrated vulnerability

For quite some time now, DoD information systems have been highly distributed. They involve a number of computers (“nodes”) connected by a routing infrastructure consisting of physical links and routers. Each computing node may play a role in more than one system; the routing infrastructure almost always supports more than one system.

Such distributed systems are now being integrated into larger scale “systems of systems”, for example “sensor to shooter” systems which integrate distributed observational systems (consisting of many planes, satellites and dozens of computers) with distributed fire control systems.

The technological reasons for this high degree of integration are powerful; the synergy obtainable is impressive. However, each element in such a complex (be it a node, a link or a router) participates in many systems and inherits vulnerabilities from each of them. The shared infrastructure has become the source of “integrated vulnerability”.



Information Survivability

Subprogram Relationships

Survivability of Large Scale Systems

intrusion detection

"public health" network for system of systems

variability & decoys

adaptable architectures & Software Engineering for system of systems

robust & secure systems

robust & secure network services

access control
fault tolerance

High Confidence Computing

High Confidence Networks

key management
robust protocols

trustworthy components

formal methods to reason about compositions of components

trustworthy components

Assurance & Integration

wrapper generation to into high confidence

transform Legacy or COTS components

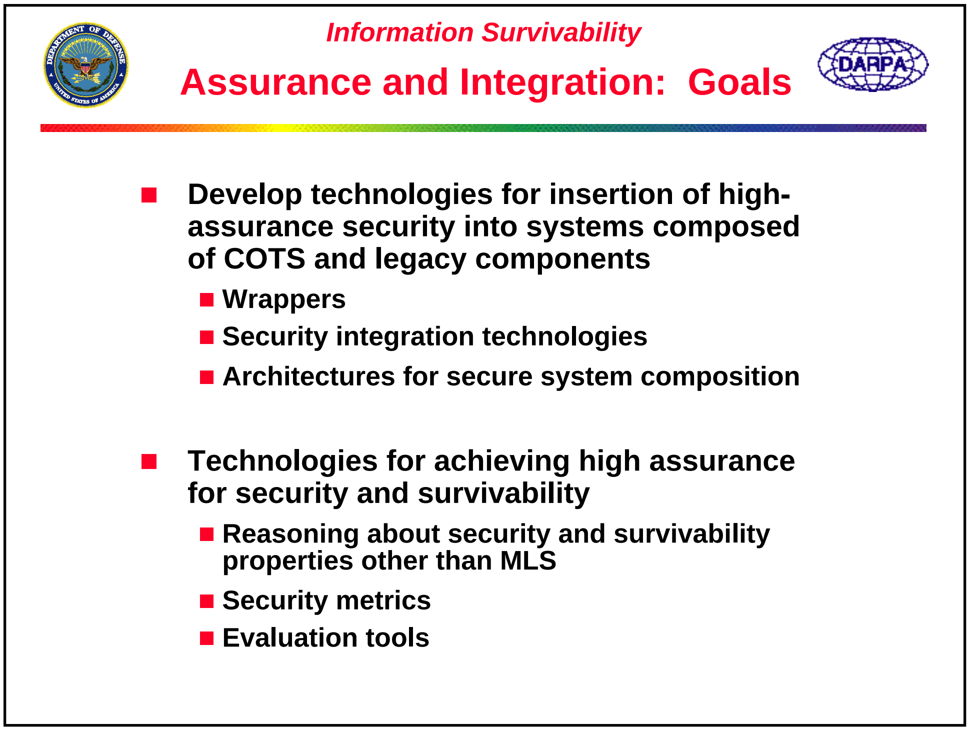
COTS

Our approach has four programmatic elements:

The **Assurance and Integration** element is responsible for developing "wrapper technology", formal reasoning methods and compositional techniques. A Wrapper is a small body of new code placed around a legacy component to guarantee that the component satisfies a design constraint (e.g. multi-level information segregation). Compositional techniques develop a set of rules for how components within a large systems may interact. Formal methods allow us to mathematically prove that the overall system will have certain desired properties if we start with components with certain properties and if we compose them according to the composition rules.

The **High Confidence Computing** and **High Confidence Networking** elements will develop techniques that assemble trustworthy components into robust computing systems and networks. It is in these program elements that we will develop technologies for access control, authentication and robustness.

The **Survivability of Large Scale Systems** element will develop technology that guarantees survivability at the large scale. This involves understanding and responding to and avoiding attacks and techniques for guaranteeing the availability of resources for critical tasks.



Information Survivability

Assurance and Integration: Goals

- **Develop technologies for insertion of high-assurance security into systems composed of COTS and legacy components**
  - **Wrappers**
  - **Security integration technologies**
  - **Architectures for secure system composition**
- **Technologies for achieving high assurance for security and survivability**
  - **Reasoning about security and survivability properties other than MLS**
  - **Security metrics**
  - **Evaluation tools**

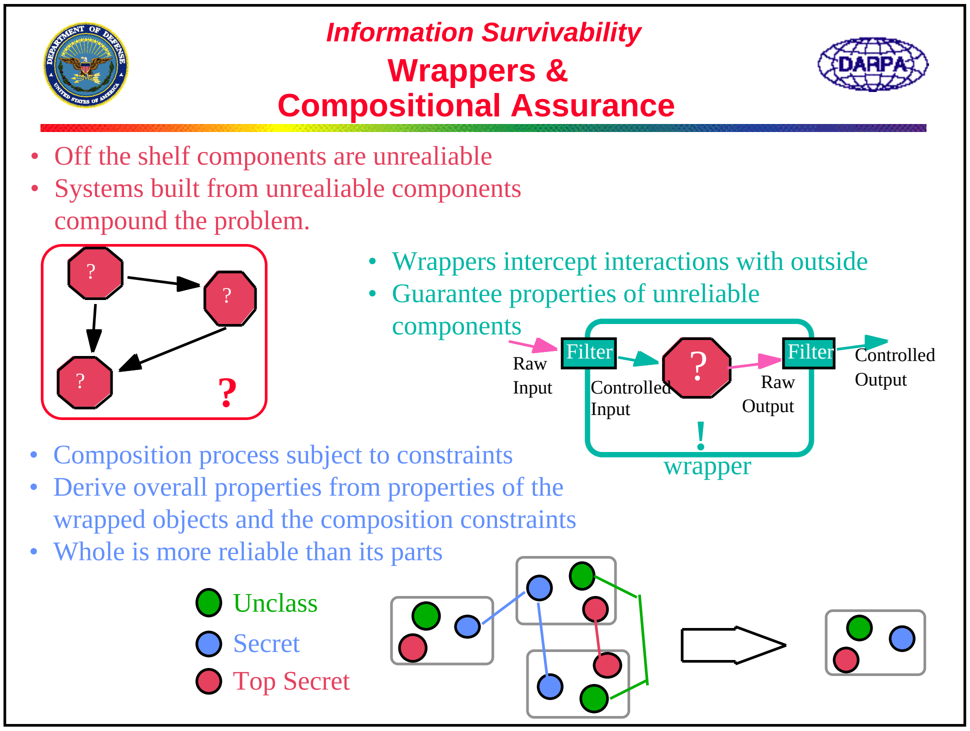
Most systems of importance today are composed of off the shelf legacy code. We therefore require ways to inject security and fault tolerance concerns into these systems so that they may be used as high confidence building blocks in the construction of tomorrow's distributed system of systems.

One promising approach is to develop a systematic "wrapper". In this approach, the existing components are surrounded by new code which guarantees those properties, something of great importance for high confidence. The Assurance and integration component of the program will work on implementing a set of security and fault tolerance techniques as wrapper components. In addition, we aim to characterize these techniques by strength and cost so that they may be used in a plug & play manner.

The second major thrust is to develop a theory of secure composition and tools for inferring system-level properties from properties of the components. Wrappers guarantee properties locally, the composition methodology will do so globally.

The third major thrust is to integrate these techniques into a software engineering methodology. This will include defining code-level security metrics and evaluation tools for evaluating the strength of systems against attack, developing tools for securely refining an abstract security architecture into a concrete system. Finally, we will

integrate these techniques into software engineering tools and apply these tools to secure, fault tolerant operating system and network services.



Information Survivability

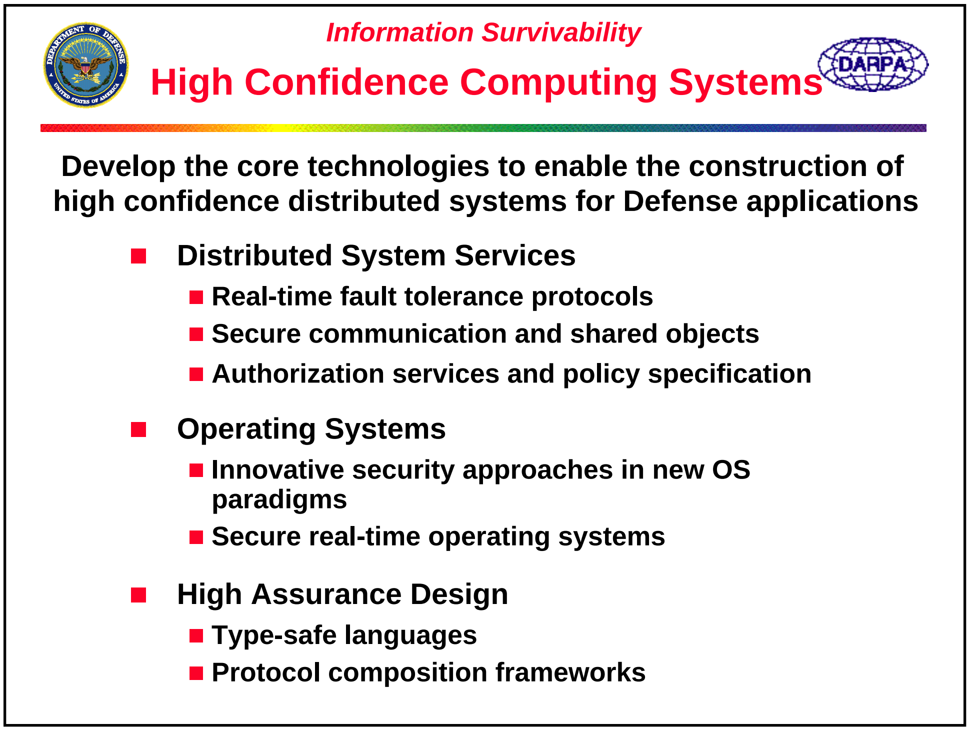
Wrappers & Compositional Assurance

- Off the shelf components are unrealiable
- Systems built from unrealiable components compound the problem.

- Wrappers intercept interactions with outside
- Guarantee properties of unreliable components

- Composition process subject to constraints
- Derive overall properties from properties of the wrapped objects and the composition constraints
- Whole is more reliable than its parts

🟢 Unclass
🔵 Secret
🔴 Top Secret



Information Survivability

High Confidence Computing Systems

Develop the core technologies to enable the construction of high confidence distributed systems for Defense applications

- **Distributed System Services**
  - **Real-time fault tolerance protocols**
  - **Secure communication and shared objects**
  - **Authorization services and policy specification**
- **Operating Systems**
  - **Innovative security approaches in new OS paradigms**
  - **Secure real-time operating systems**
- **High Assurance Design**
  - **Type-safe languages**
  - **Protocol composition frameworks**

The High Confidence Computing Systems (HCC) component of the Information Survivability program is developing the core technologies necessary for construction of a high confidence distributed computing base. The term “High Confidence” encompasses the critical system properties of security, dependability (or fault-tolerance), and real-time responsiveness.

HCC is coupled to several other ITO programs such as Global Mobile Computing, Embeddable Systems, and Quorum to ensure that the core technologies are developed and validated in a larger system context. HCC also jointly supports projects with NSA, Rome Lab, and NSF.

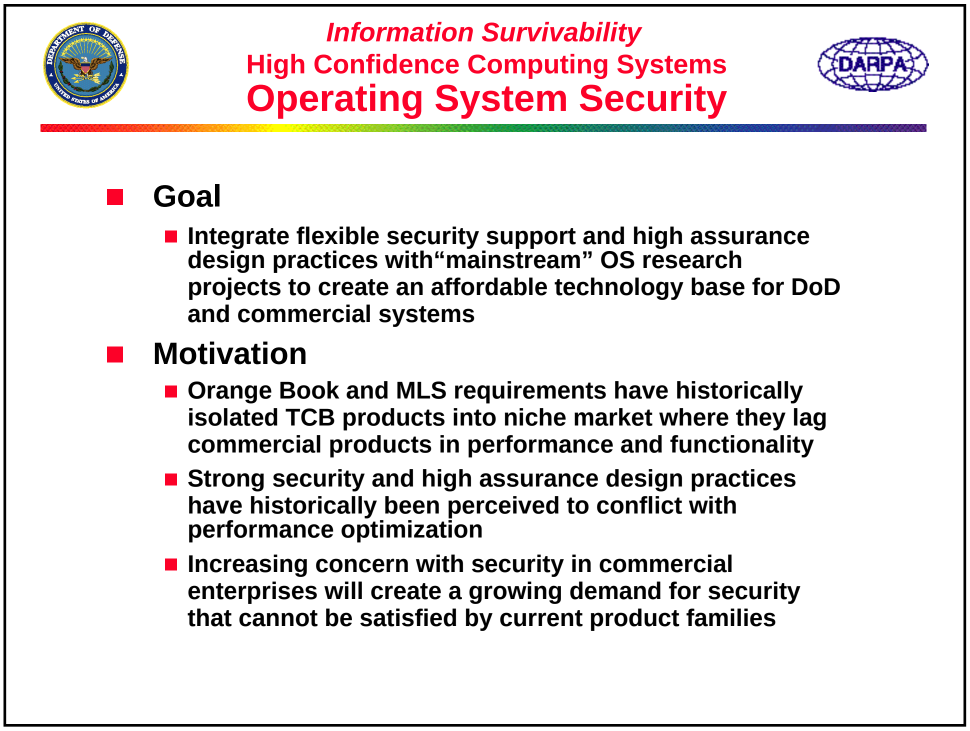
HCC projects may be broadly classified as addressing technology needs primarily at the Distributed Systems level, the Operating System level, or the Design Assurance level.

A key challenge at all levels is the development of integrated approaches for ensuring real-time, dependability, and security properties. Thus new protocols for achieving fault-tolerance under real-time constraints and secure real-time operating systems are being developed.

Emerging distributed environments pose new challenges to security. New operating system paradigms are being explored in the Quorum program that provide higher performance in such environments through provisions for application-level resource management and control, such as extensible kernels, and dynamic code generation. HCC is forging collaborations between the operating systems and security communities to identify innovative approaches to addressing security issues in these promising new designs.

HCC is also developing technologies for authorization and enforcement of access control policies relevant to new distributed computing paradigms (work flow systems and shared objects).

Finally, HCC is developing powerful methods for the high assurance design of systems software.



Information Survivability

High Confidence Computing Systems

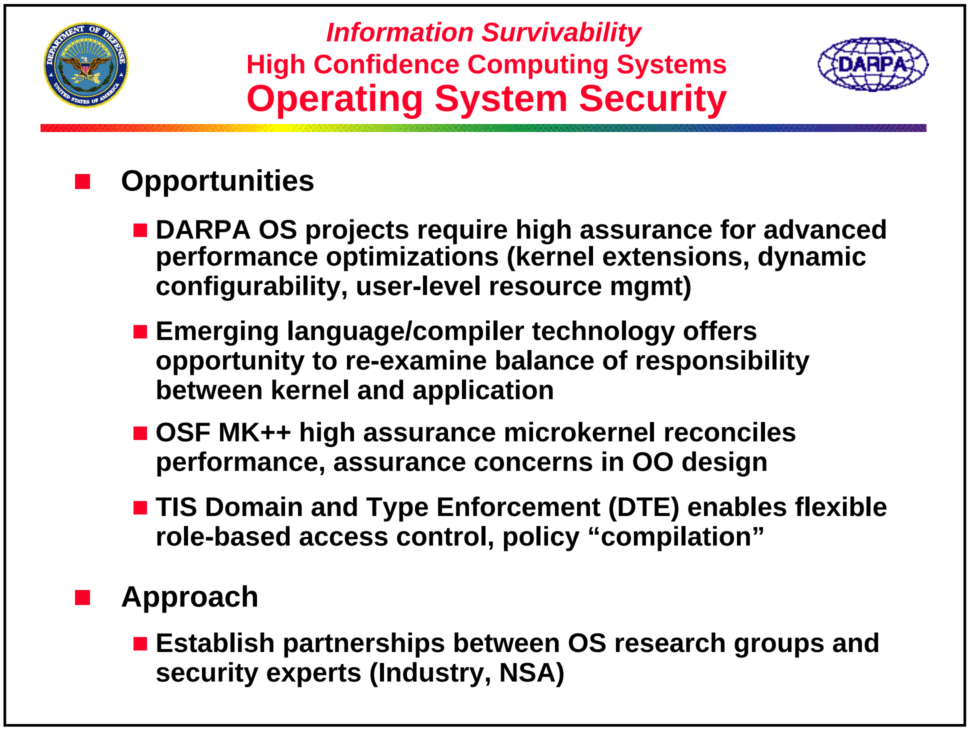
Operating System Security

- Goal
  - Integrate flexible security support and high assurance design practices with“mainstream” OS research projects to create an affordable technology base for DoD and commercial systems
- Motivation
  - Orange Book and MLS requirements have historically isolated TCB products into niche market where they lag commercial products in performance and functionality
  - Strong security and high assurance design practices have historically been perceived to conflict with performance optimization
  - Increasing concern with security in commercial enterprises will create a growing demand for security that cannot be satisfied by current product families

The goal of the Operating System Security thrust of the High Confidence Computing Systems program is to integrate security awareness, flexible mechanisms, and high assurance design practices into “mainstream” operating system research projects which are primarily focused on innovative approaches to achieving high performance. Since the results of these projects are likely to impact future commercial products it is imperative that adequate support for DoD security requirements be an integral part of their architecture, design, and implementation in order to ensure a commercially sustainable, affordable technology base for Defense systems.

Historically, secure operating system products have occupied a niche Defense market separate from the much larger commercial one. Driven by the strict demands of the Trusted Computing System Evaluation Criteria (Orange Book), the multi-level security (MLS) model, and the principle of the Trusted Computing Base (TCB), these products have been characterized by long design and certification times causing them to significantly lag commercial operating systems in performance, functionality and usability. Performance-oriented commercial designs have often sacrificed assurance and security support to meet performance requirements and time-to-market demands. There is a strong perception that performance and security requirements conflict with one another and the market has become divided between the two.

The growing visibility of security concerns in commercial enterprises triggered, in part, by increased networking and distribution of resources and, in part, by the desire to enforce organization-specific policies will create an increasing demand for operating systems that provide both high performance and flexible, high assurance security. Existing technologies, such as firewalls and commercial operating systems, are unable to satisfy this demand and the historic isolation of security research from commercial practice leaves industry poorly positioned to address the problem.



Information Survivability

High Confidence Computing Systems

Operating System Security

- **Opportunities**
  - **DARPA OS projects require high assurance for advanced performance optimizations (kernel extensions, dynamic configurability, user-level resource mgmt)**
  - **Emerging language/compiler technology offers opportunity to re-examine balance of responsibility between kernel and application**
  - **OSF MK++ high assurance microkernel reconciles performance, assurance concerns in OO design**
  - **TIS Domain and Type Enforcement (DTE) enables flexible role-based access control, policy "compilation"**
- **Approach**
  - **Establish partnerships between OS research groups and security experts (Industry, NSA)**

Recent results from DARPA-sponsored research have created a technology foundation and unprecedented opportunity for integrated attacks on security and performance issues in operating systems research.

Current DARPA-sponsored OS projects are exploring innovative approaches to achieving high performance through smaller or more flexible kernel architectures. Methods include kernel extensions, dynamic configurability,dynamic code generation, and enhanced user-level resource management. Effective implementation of these concepts has generated renewed interest in issues of safety and assurance in the OS community--the foundation of security doctrine. The OS community has exploited advances in language and compiler technology such as optimized type-safe languages, partial evaluation, software fault isolation to achieve assurance. The potential of this approach remains largely untapped and its limitations have not yet been adequately explored by security experts. If viable, language and compiler technology opens up the possibility of redefining security as a shared application/OS responsibility enabling performance optimizations and overcoming the rigidity of the kernel-centric Orange Book view.

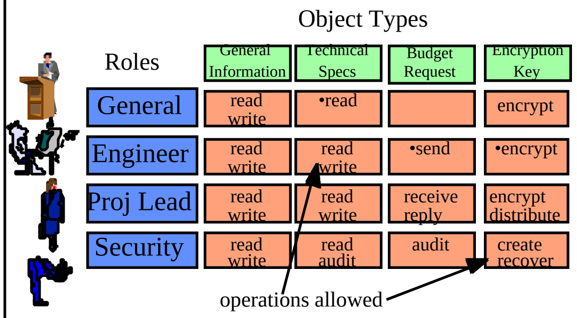
The Orange Book emphasizes high assurance design. The embrace of object-oriented design methods by the OS community is a step in this direction but conventional OO design practice, while enhancing manageability through modularization, does not necessarily equate to high assurance. OSF has demonstrated, however, that OO design can be used to support both high performance and high assurance compatible with Orange Book B3 criteria.

Finally, the Domain and Type Enforcement technology from Trusted Information Systems has demonstrated the viability of supporting a highly flexible organization-specific policy in a commercial operating system.

This convergence of technologies and concerns has created opportunities for cooperative research in support of DARPA and NSA objectives.

Information Survivability

High Confidence Computing
Role Based Access Control

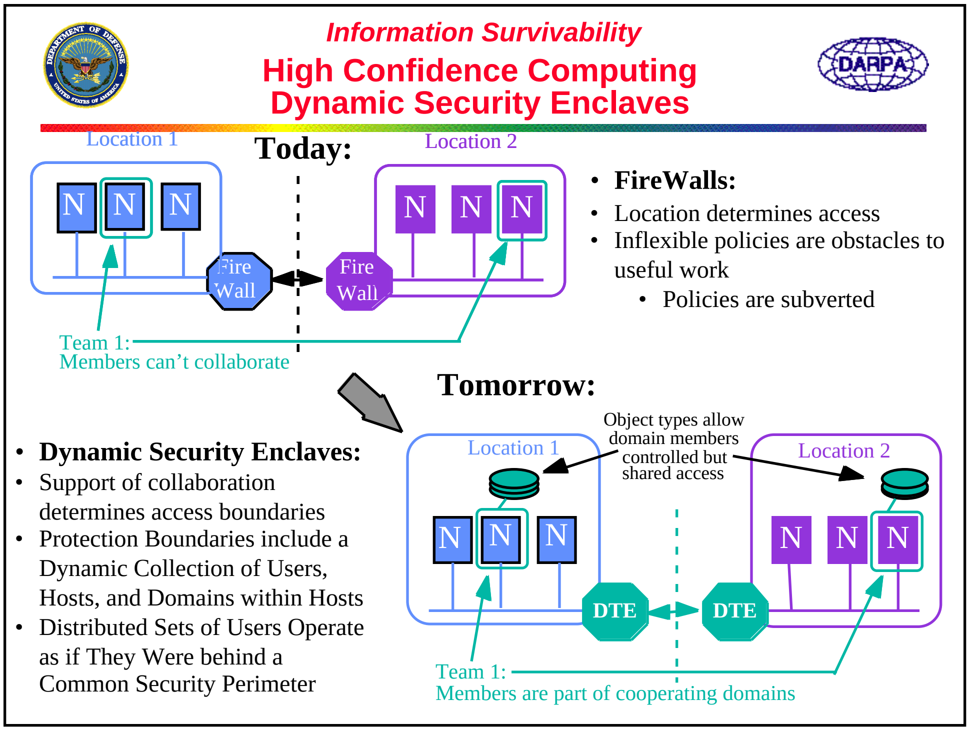


Object Types

Roles	General Information	Technical Specs	Budget Request	Encryption Key
General	read write	•read		encrypt
Engineer	read write	read write	•send	•encrypt
Proj Lead	read write	read write	receive reply	encrypt distribute
Security	read write	read audit	audit	create recover

- Access is controlled by the organizational role a user (or system process) is playing.
- Users and/or processes must authenticate their right to assume a particular role at a particular time.
  - Dynamic and Static Constraints govern Role assignment.
- Resources (files, network streams) are assigned types.
- Role and Type determines access to the operation available on an object..

- Highly flexible policies are possible
  - Extend taxonomy of roles, types & operations to give needed expressive power.
  - No process need be given greater authority than it actually needs to achieve its goals.
- Organizations may design their own policy.
  - expressed in terms of roles, object types and allowed operations.
  - Compiled into lower level enforcment mechanisms provided by OS.
- Security concerns need not be a obstacle in the way of getting legitimate work done.



Information Survivability

High Confidence Computing
Dynamic Security Enclaves

Today:

Location 1

N N N

Fire Wall

Fire Wall

Location 2

N N N

Team 1:
Members can’t collaborate

- **FireWalls:**
- Location determines access
- Inflexible policies are obstacles to useful work
  - Policies are subverted

Tomorrow:

Location 1

N N N

DTE

Object types allow
domain members
controlled but
shared access

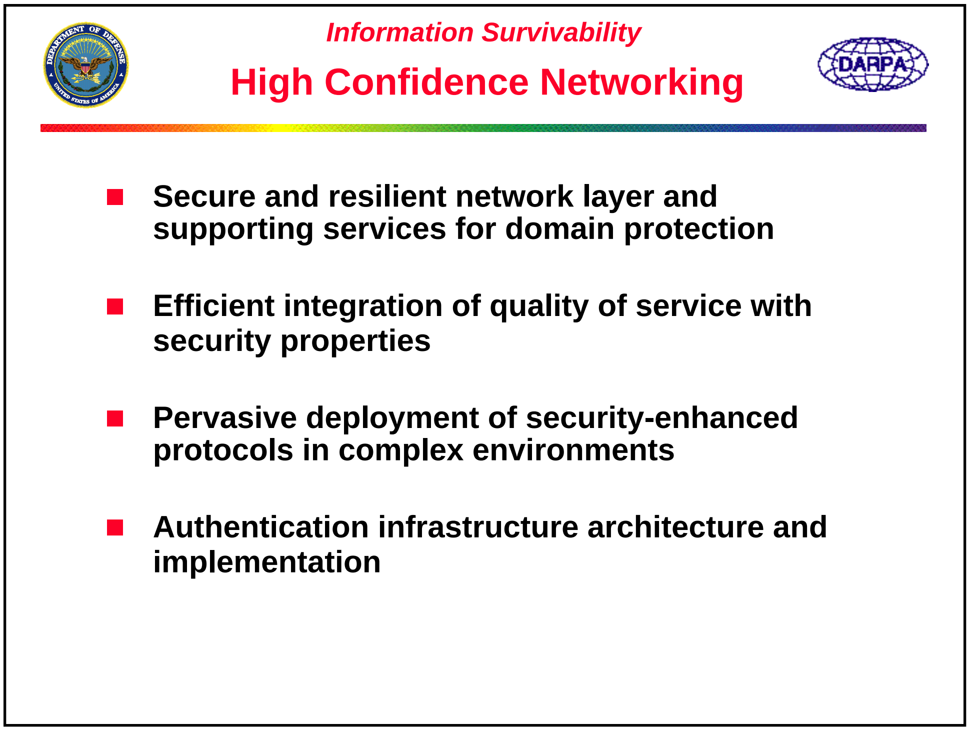
DTE

Location 2

N N N

Team 1:
Members are part of cooperating domains

- **Dynamic Security Enclaves:**
- Support of collaboration determines access boundaries
- Protection Boundaries include a Dynamic Collection of Users, Hosts, and Domains within Hosts
- Distributed Sets of Users Operate as if They Were behind a Common Security Perimeter



Information Survivability

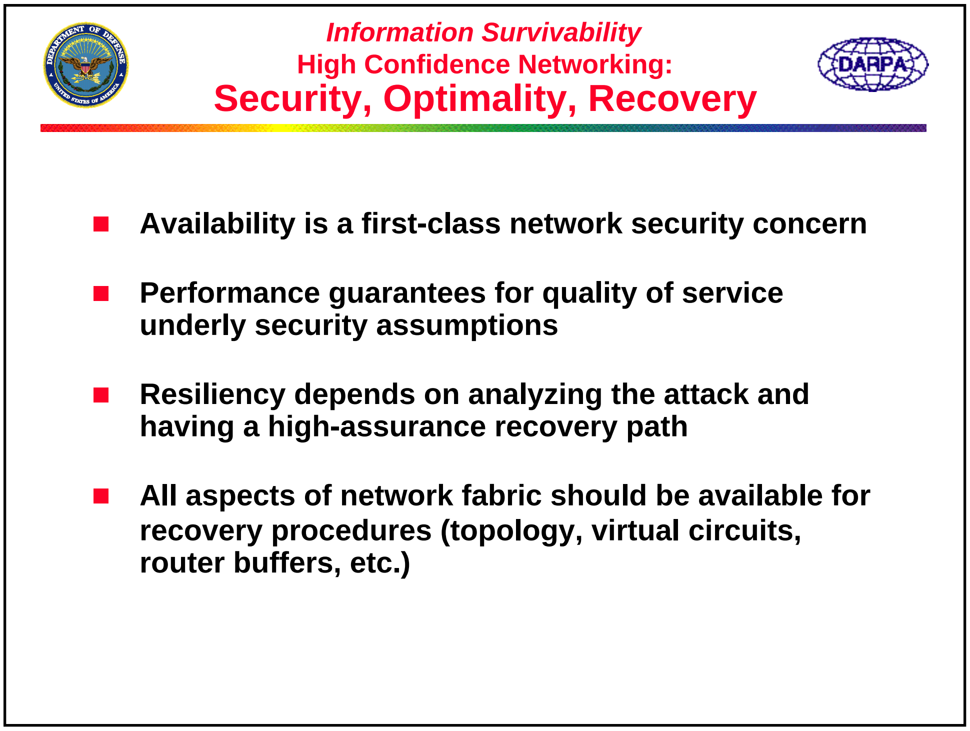
High Confidence Networking

- Secure and resilient network layer and supporting services for domain protection
- Efficient integration of quality of service with security properties
- Pervasive deployment of security-enhanced protocols in complex environments
- Authentication infrastructure architecture and implementation

The HCN area builds the software infrastructure for binding together networks into secure and resilient communications substrates. The security is founded on the concepts of authentication, confidentiality, and integrity of the of the network infrastructure protocols and routers, and the resiliency is founded on the ability of the network to manage resources, detect service-threatening events, and to utilize redundancy to protect and restore resources.

The challenge facing secure network researchers is to bind the multitude of security mechanisms that have been recently developed into a smoothly functioning dynamic system that is responsive and manageable. Users have increasingly sophisticated demands for secure communication services, and the network services must be available to meet the demands efficiently, even under stress.

The HCN architecture will be able to provision desired security services with maximum efficiency, even in complex environments with multiple technologies for transport, cryptography, and firewalls, mobile nodes, mobile code, etc..



Information Survivability

High Confidence Networking:

Security, Optimality, Recovery

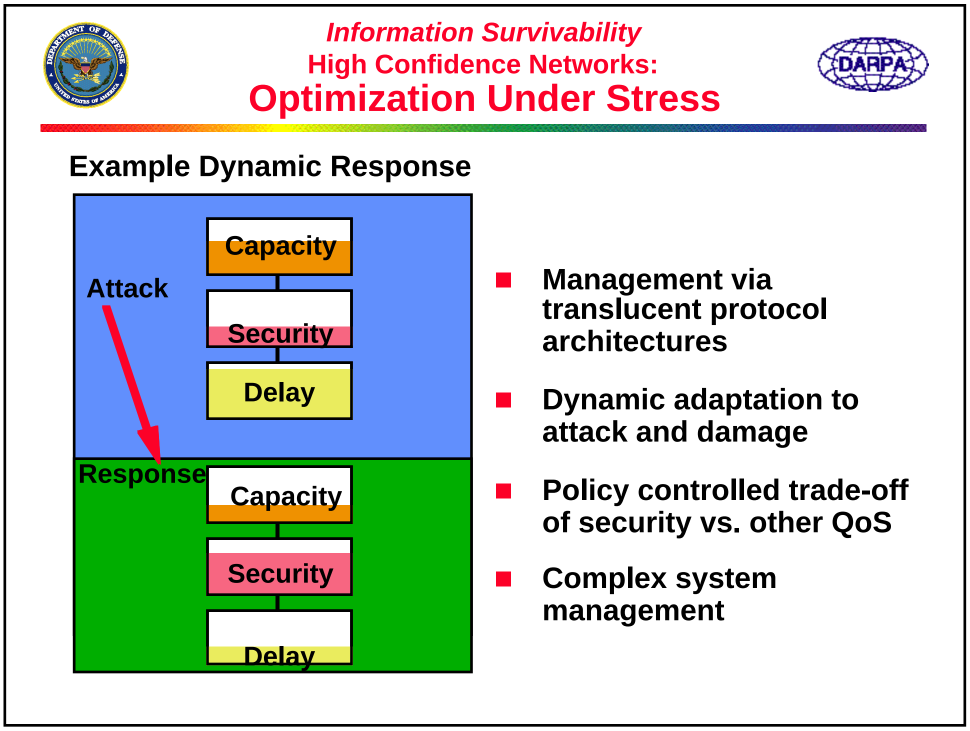
- Availability is a first-class network security concern
- Performance guarantees for quality of service underly security assumptions
- Resiliency depends on analyzing the attack and having a high-assurance recovery path
- All aspects of network fabric should be available for recovery procedures (topology, virtual circuits, router buffers, etc.)

High Confidence Networks must be able to continue functioning even when resources are dimished or damaged, but the ability to recover can be compromised if the network capacity is reduced. This has motivated the inclusion of availability and dependability as crucial aspects of an HCN.

Another dimension in thedesign of networks is the inclusion of quality of service guarantees such as priority, delay, jitter, redundancy, etc. Security attributes will also be part of the quality of service in HCN’s, but security itself is dependent on availability.

The challenge in runtime management of HCN’s is to optimize the use of the resources that remain during an attack so that recovery can proceed as rapidly and securely as possible. This is similar to normal network management, but with a different emphasis on priority, safety, and security.

All aspects of the network fabric must be utilized for analyzing an attack on the physical or logical infrastructure, and all elements must be available for managing an optimal recovery. Assuring the communication of the information and the management commands over a damaged structure is the fundamental problem posed to network designers.



Information Survivability

High Confidence Networks:

Optimization Under Stress

Example Dynamic Response

Capacity
Attack
Security
Delay
Response
Capacity
Security
Delay

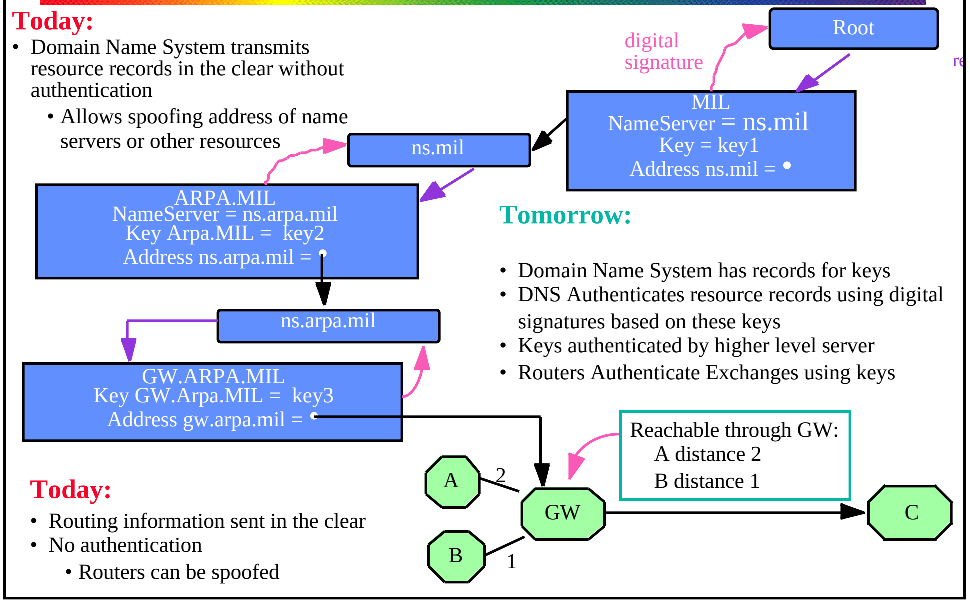
- Management via translucent protocol architectures
- Dynamic adaptation to attack and damage
- Policy controlled trade-off of security vs. other QoS
- Complex system management

A flexible and malleable protocol structure can facilitate the inclusion of repair and recovery mechanisms in a HCN. The diagram above suggests how the network components that guarantee various quality of service parameters might be directed to respond during an attack. Connections crucial to continuing operation may have their link capacity reduced while increasing the security and decreasing delay. The protocol structure that achieves this adaptation includes sharing information between several different levels of network organization and protocol layers, illustrating the need to expand the capabilities of network protocol architectures.

The management of the adaptation depends on prior analysis of threats, high assurance software, and policies that define the allowable trade-offs between the parameters, including the various components of security. The overall management problem is challenging, and the ability to design, manage, and evolve increasingly complicated systems will be the hallmark of 21st century system engineers.

Information Survivability

High Confidence Networking
Securing the Infrastructure



Today:

- Domain Name System transmits resource records in the clear without authentication
  - Allows spoofing address of name servers or other resources

digital
signature

Root

re

MIL
NameServer = ns.mil
Key = key1
Address ns.mil = •

ns.mil

ARPA.MIL
NameServer = ns.arpa.mil
Key Arpa.MIL = key2
Address ns.arpa.mil = •

ns.arpa.mil

GW.ARPA.MIL
Key GW.Arpa.MIL = key3
Address gw.arpa.mil = •

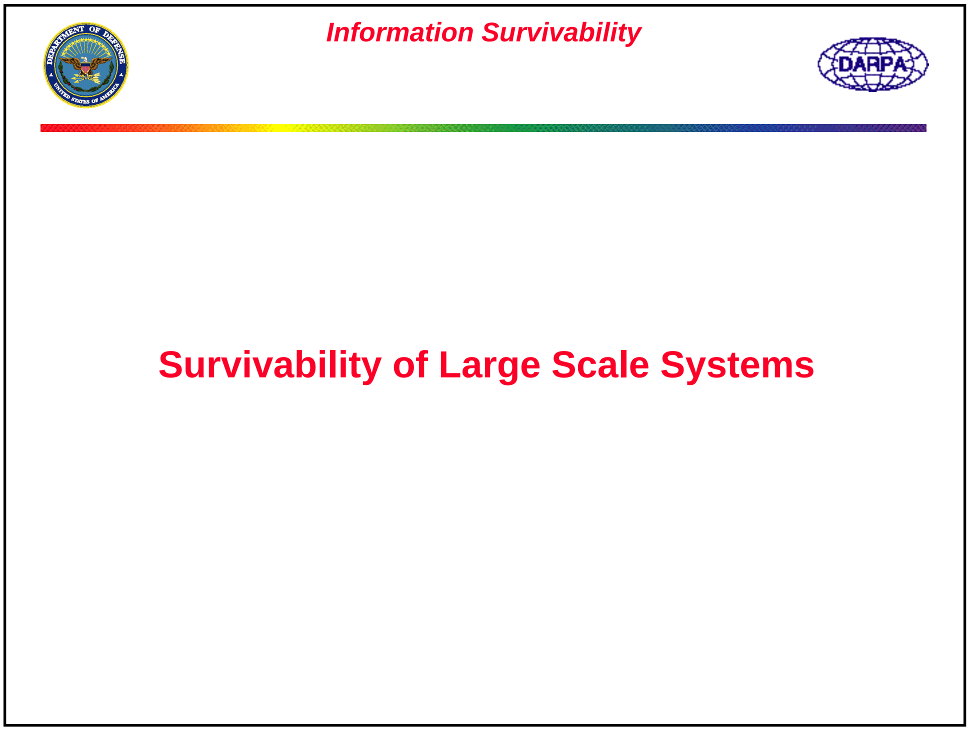
Tomorrow:

- Domain Name System has records for keys
- DNS Authenticates resource records using digital signatures based on these keys
- Keys authenticated by higher level server
- Routers Authenticate Exchanges using keys

Reachable through GW:
A distance 2
B distance 1

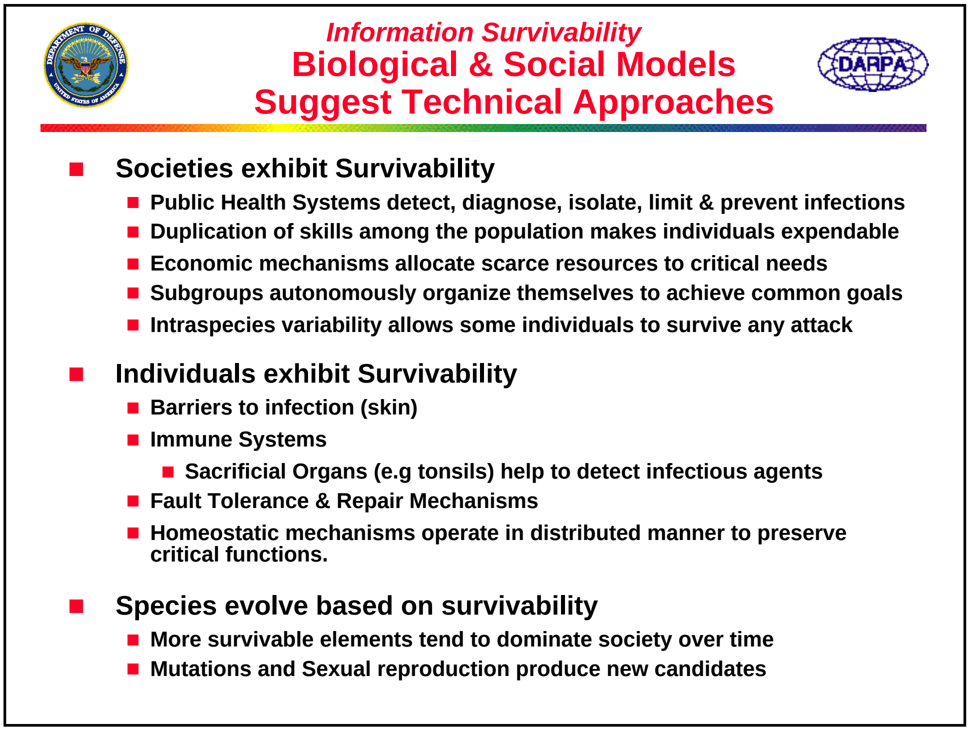
Today:

- Routing information sent in the clear
- No authentication
  - Routers can be spoofed



Information Survivability

Survivability of Large Scale Systems



Information Survivability

Biological & Social Models Suggest Technical Approaches

- Societies exhibit Survivability
  - Public Health Systems detect, diagnose, isolate, limit & prevent infections
  - Duplication of skills among the population makes individuals expendable
  - Economic mechanisms allocate scarce resources to critical needs
  - Subgroups autonomously organize themselves to achieve common goals
  - Intraspecies variability allows some individuals to survive any attack
- Individuals exhibit Survivability
  - Barriers to infection (skin)
  - Immune Systems
    - Sacrificial Organs (e.g tonsils) help to detect infectious agents
  - Fault Tolerance & Repair Mechanisms
  - Homeostatic mechanisms operate in distributed manner to preserve critical functions.
- Species evolve based on survivability
  - More survivable elements tend to dominate society over time
  - Mutations and Sexual reproduction produce new candidates

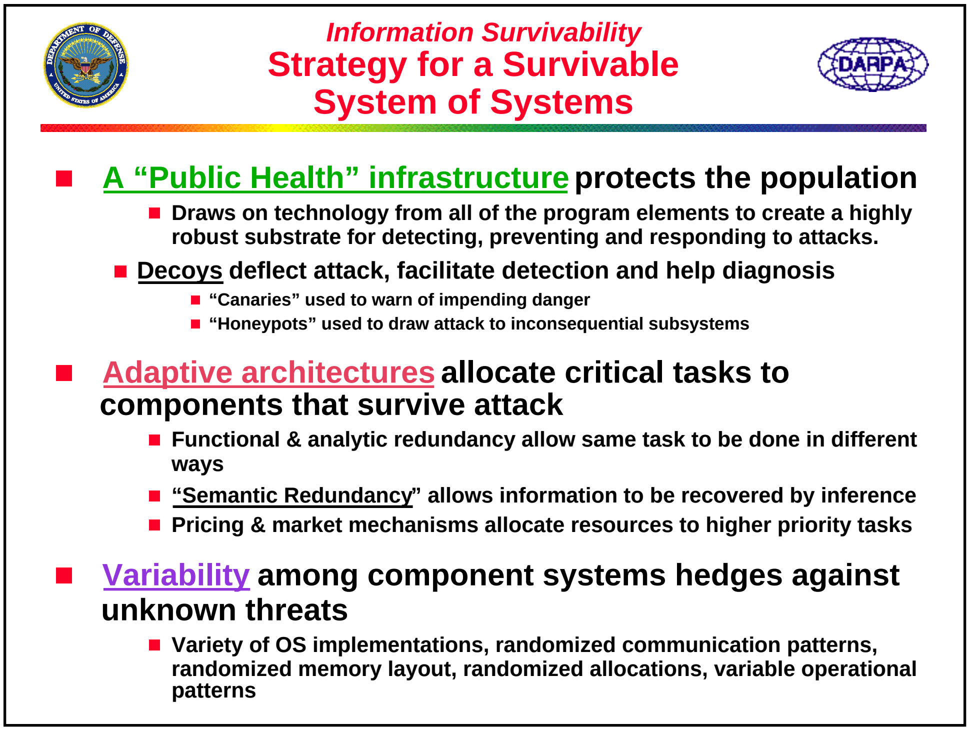
In looking for guidance about how to design survivable systems we have chosen to look at the naturally occurring models of survivability: biological organisms and populations or societies.

Individual organisms have barriers to infection and immune systems which both detect the presence of infections and mount a counter-attack to remove the foreign agents. Organisms also involve a degree of redundancy and fault tolerance that isn't normally seem in the information systems that we design. Finally, biological systems have homeostatic mechanisms that guarantee that critical functions are maintained even when the organism is under stress.

These same ideas are reproduced at the macro scale by populations and societies. Public health systems deal with infections, economies deal with allocation of scarce resources to critical needs. From the population's viewpoint the individual organisms are redundant elements; the population as a whole is fault tolerant to the loss of individuals.

However, the population as a whole has an additional survival strategy which is variability among the population: any particular attack will find some elements of the population immune to it and these immune elements form the basis for reconstituting the society. The species as a whole evolves based on which individuals prove to be survivable in the environment.

The challenge is to figure out how to apply this to information systems!



Information Survivability

Strategy for a Survivable System of Systems

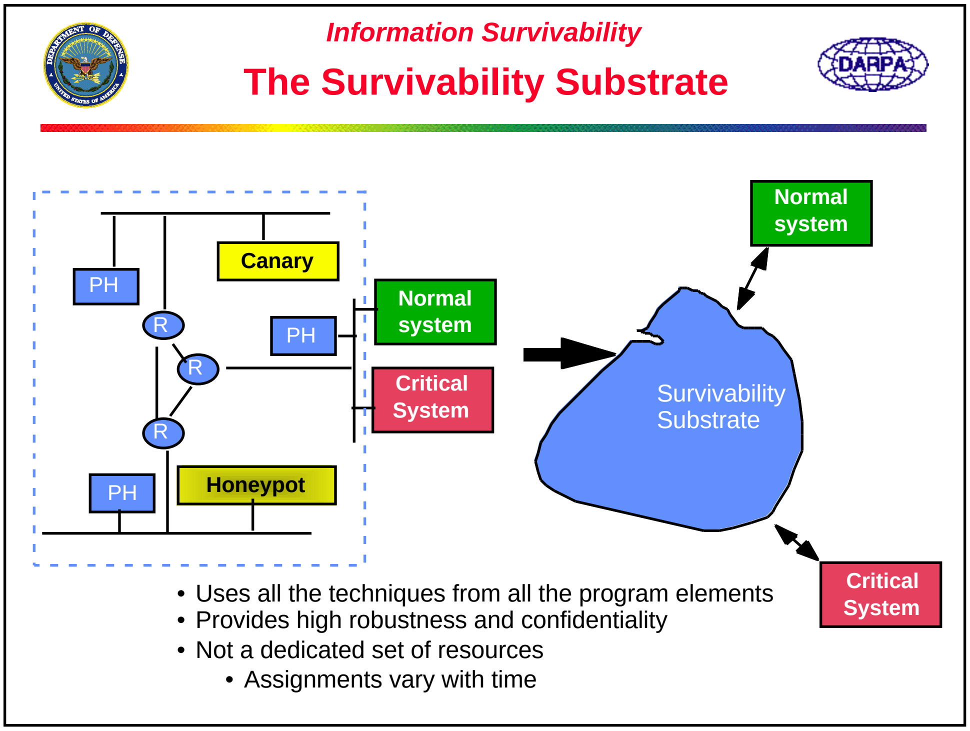
- **A “Public Health” infrastructure protects the population**
  - **Draws on technology from all of the program elements to create a highly robust substrate for detecting, preventing and responding to attacks.**
  - **Decoys deflect attack, facilitate detection and help diagnosis**
    - **“Canaries” used to warn of impending danger**
    - **“Honeypots” used to draw attack to inconsequential subsystems**
- **Adaptive architectures allocate critical tasks to components that survive attack**
  - **Functional & analytic redundancy allow same task to be done in different ways**
  - **“Semantic Redundancy” allows information to be recovered by inference**
  - **Pricing & market mechanisms allocate resources to higher priority tasks**
- **Variability among component systems hedges against unknown threats**
  - **Variety of OS implementations, randomized communication patterns, randomized memory layout, randomized allocations, variable operational patterns**

Our research on Survivability of Large Scale Systems will attempt to develop engineering methodology for large scale DoD systems of systems using these observations as models. In particular, the research will concentrate in three areas:

The **Public Health Infrastructure** element will draw on technologies developed in all elements of the program to develop a highly survivable infrastructure. This isn’t a dedicated set of links and nodes, but rather a set of security and survivability protocols that can operate with whatever resources are available. This infrastructure will be used to support collaborative problem solving between components of the system to in detecting, understanding and responding to attacks.

The **Adaptive Architecture** element will develop technologies that guarantee resources to critical tasks and that allow the system to continue functioning even when successful attacks have been mounted. For example, pricing mechanisms can be used to reflect the availability of resources (or lack thereof); in such a model applications of lower priority will not be able to obtain expensive critical resources, because their budget will be too small. In addition, this element will attempt to develop new models of redundancy that will allow corrupted data to be recovered by inference.

The **Variability** element will develop technologies that allow us to introduce differences between individual systems even when they are running common (COTS) software.



Information Survivability

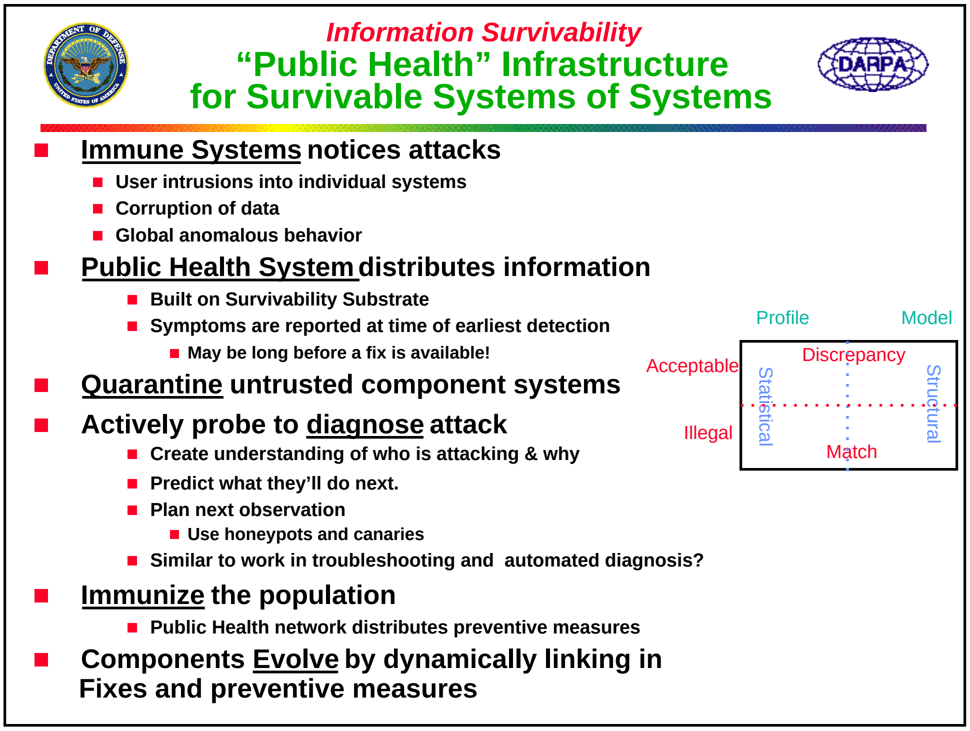
The Survivability Substrate

- Uses all the techniques from all the program elements
- Provides high robustness and confidentiality
- Not a dedicated set of resources
  - Assignments vary with time

To make such an approach work we will need a highly survivable and trustworthy substrate which can support core functions. This “**Survivability Substrate”** isn’t a dedicated set of resources; this would only be a visible target for attack. Instead it is an adaptive set of protocols using all the technologies we are developing. These protocols will find and use whatever resources are available; they will avoid centralized control whenever possible so as to avoid a single point of vulnerability. Furthermore, as we will see later on, the functions played by the survivability substrate will be provided by a time varying set of physical resources; such variability will make it harder for an adversary to understand what to attack.

Normal systems, highly critical systems, and elements of the public health function itself will use this survivability substrate to communicate information when necessary.

The survivability substrate may at various times use certain elements as decoys: **Canaries** are normal systems without special protections; like a miners canary, if these systems start to malfunction it is probably a sign of impending trouble. **Honeypots**, in contrast, are systems designed to look important and highly instrumented to detect not only the presence of attacks but also the means and method of attack.



Information Survivability

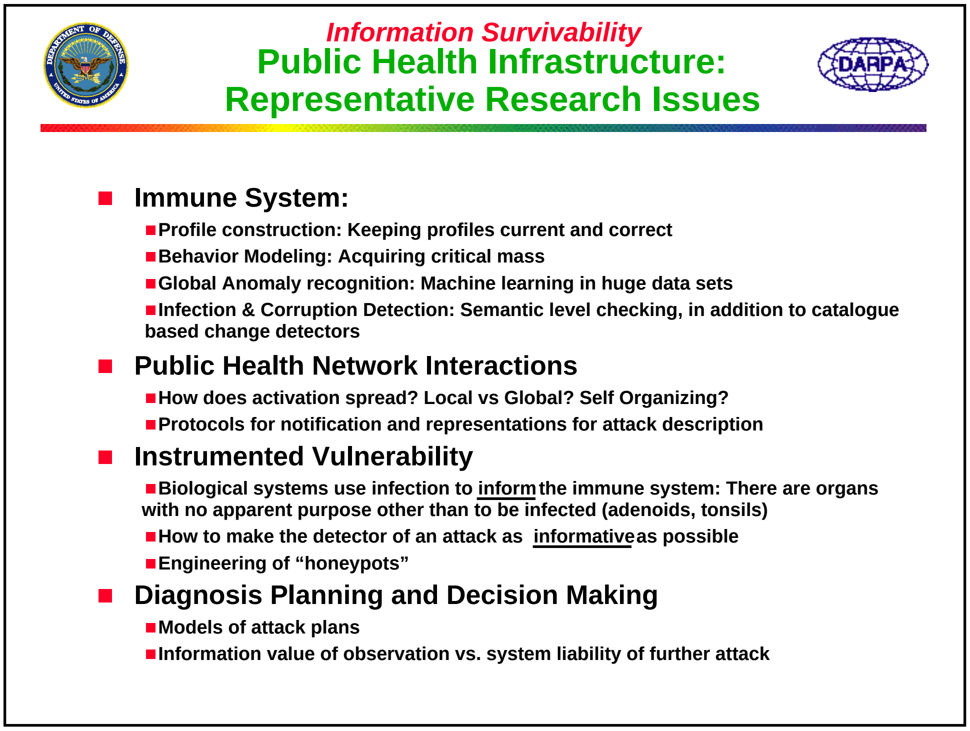
"Public Health" Infrastructure for Survivable Systems of Systems

- Immune Systems notices attacks
  - User intrusions into individual systems
  - Corruption of data
  - Global anomalous behavior
- Public Health System distributes information
  - Built on Survivability Substrate
  - Symptoms are reported at time of earliest detection
    - May be long before a fix is available!
- Quarantine untrusted component systems
- Actively probe to diagnose attack
  - Create understanding of who is attacking & why
  - Predict what they'll do next.
  - Plan next observation
    - Use honeypots and canaries
  - Similar to work in troubleshooting and automated diagnosis?
- Immunize the population
  - Public Health network distributes preventive measures
- Components Evolve by dynamically linking in Fixes and preventive measures

One of our design goals is to build a "public health" infrastructure for DoD systems of systems. Such an infrastructure must be able to detect the symptoms of an attack at the earliest time possible. Having noticed such symptoms in some localities, it then may try to communicate this information to other localities and to involve other systems in the attempt to understand what is going on. When systems have been identified as corrupted, the infrastructure will need to "quarantine" them so that they do not corrupt other elements of the system.

As signs of attack multiply, the system will need to raise its level of concern and to draw more elements at more places into the effort to diagnose the attack. Ideally, as vulnerabilities are discovered, the infrastructure can securely relay information about these vulnerabilities to other elements of the public health system without disclosing this information to individuals who might misuse it. Also, as means are discovered to counteract and recover from elements of the attack, the public health infrastructure will spread this information.

Computing elements will dynamically link in these fixes as they are discovered, in effect, immunizing the population. This dynamic linking in of fixes is one way in which the population evolves towards a more survivable form.



Information Survivability

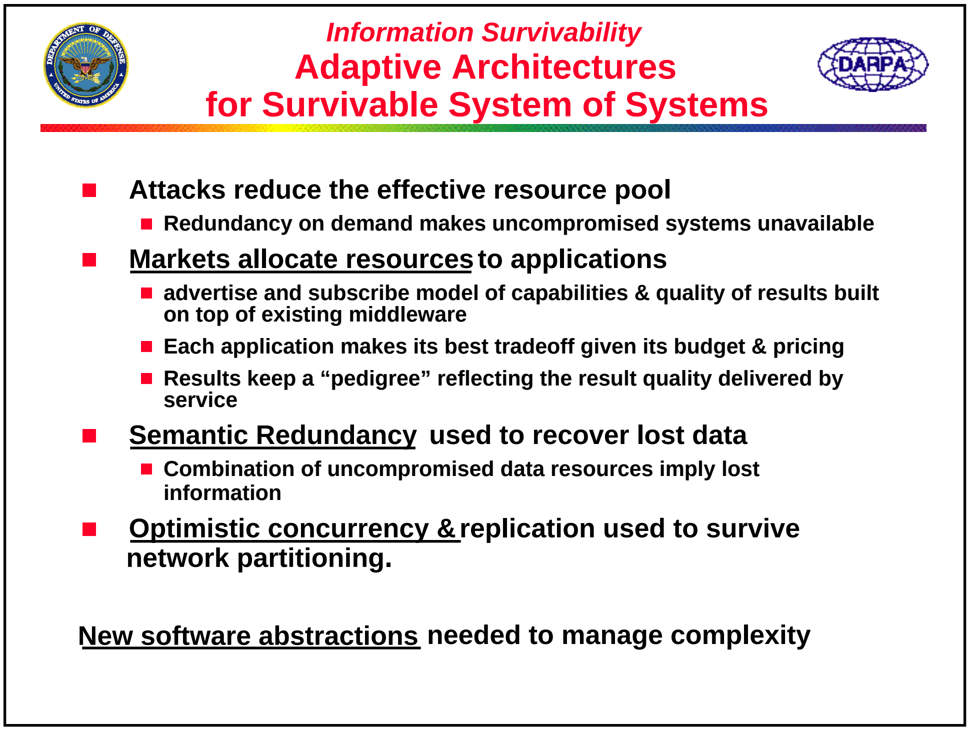
Public Health Infrastructure: Representative Research Issues

- **Immune System:**
  - **Profile construction: Keeping profiles current and correct**
  - **Behavior Modeling: Acquiring critical mass**
  - **Global Anomaly recognition: Machine learning in huge data sets**
  - **Infection & Corruption Detection: Semantic level checking, in addition to catalogue based change detectors**
- **Public Health Network Interactions**
  - **How does activation spread? Local vs Global? Self Organizing?**
  - **Protocols for notification and representations for attack description**
- **Instrumented Vulnerability**
  - **Biological systems use infection to inform the immune system: There are organs with no apparent purpose other than to be infected (adenoids, tonsils)**
  - **How to make the detector of an attack as informative as possible**
  - **Engineering of "honeypots"**
- **Diagnosis Planning and Decision Making**
  - **Models of attack plans**
  - **Information value of observation vs. system liability of further attack**

It isn't clear at this stage just how all of this is best done. To what degree is this a self organizing process and to what extent is there coordination or centralization of control. Is survivability an emergent property or one designed in?

In medicine there are times when it is better not to treat a patient until more information can be gathered; of course, waiting also entails a risk. Similarly, in trying to diagnose an attack on a large scale information system there will be times when it is better not to fix a system which has been compromised because there is possibility of learning much more about the mode of attack if the systems isn't fixed. What framework will allow us to make such decisions correctly, and how do we automate it?

As we see an attack developing, or more correctly as we see more symptoms of what might be an attack, we will need to figure out how to gather the most informative information. This may involve hypothesizing different attacks and then predicting what should be observable in each of these attacks. What kind of knowledge base and reasoning methods will support this hypothesizing and observation planning?



Information Survivability

Adaptive Architectures for Survivable System of Systems

- Attacks reduce the effective resource pool
  - Redundancy on demand makes uncompromised systems unavailable
- Markets allocate resources to applications
  - advertise and subscribe model of capabilities & quality of results built on top of existing middleware
  - Each application makes its best tradeoff given its budget & pricing
  - Results keep a “pedigree” reflecting the result quality delivered by service
- Semantic Redundancy used to recover lost data
  - Combination of uncompromised data resources imply lost information
- Optimistic concurrency & replication used to survive network partitioning.

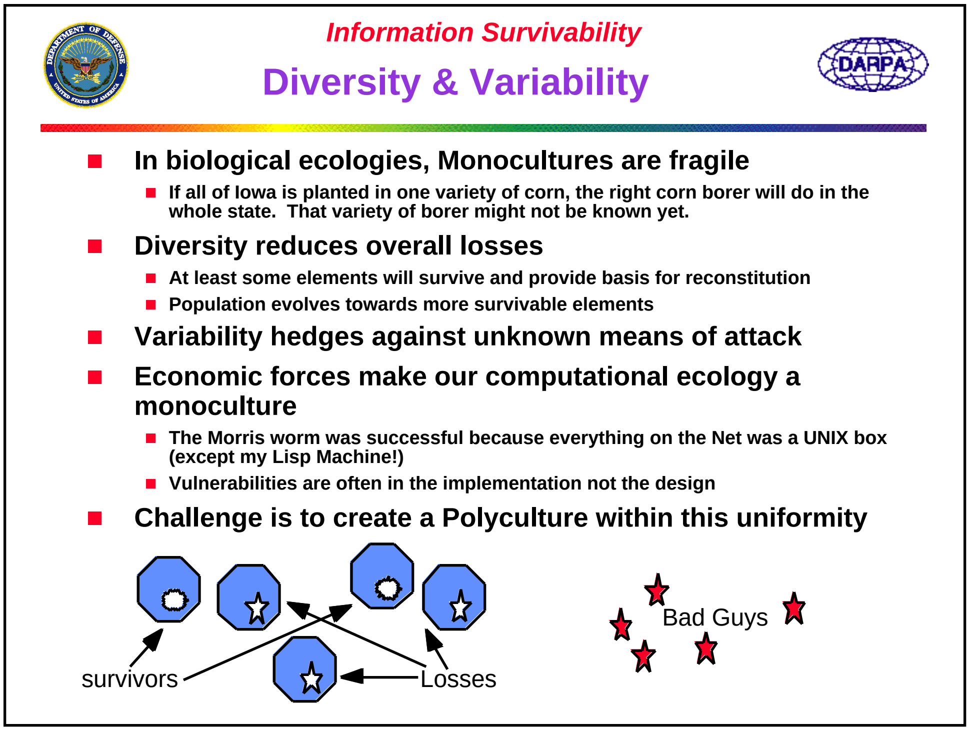
New software abstractions needed to manage complexity

When attacks are launched against the system, the set of resources available will be diminished. Even if the attack is prevented, resources will need to be diverted into performing the “public health” or “immune” functions necessary to deflect the attack; these resources are not available to get real work done as long as they are working on preventing attacks.

Our systems must have a highly adaptive architecture to guarantee that critical functions continue to be performed even as resources are lost and information is compromised.

One method for guiding resource allocation is to think of the services provided by our systems as commodities in a market economy. If we know who is demanding these services and how much of them are being provided, then we can calculate their price. By allocating larger budgets to the more critical applications, we can then guarantee that they will have the resources to buy the services they need to get their jobs done. Less critical applications will have to wait.

We also need to worry about corruption of information. In addition to the standard techniques used today for highly reliable databases, we will also investigate whether there are ways to engineer “semantic redundancy” into related databases; semantic redundancy occurs when the contents of one database implies the contents of other databases. Such redundancy is a good thing, it allows us to use one database as a check on the other and to recover lost data in one by inference.



Information Survivability

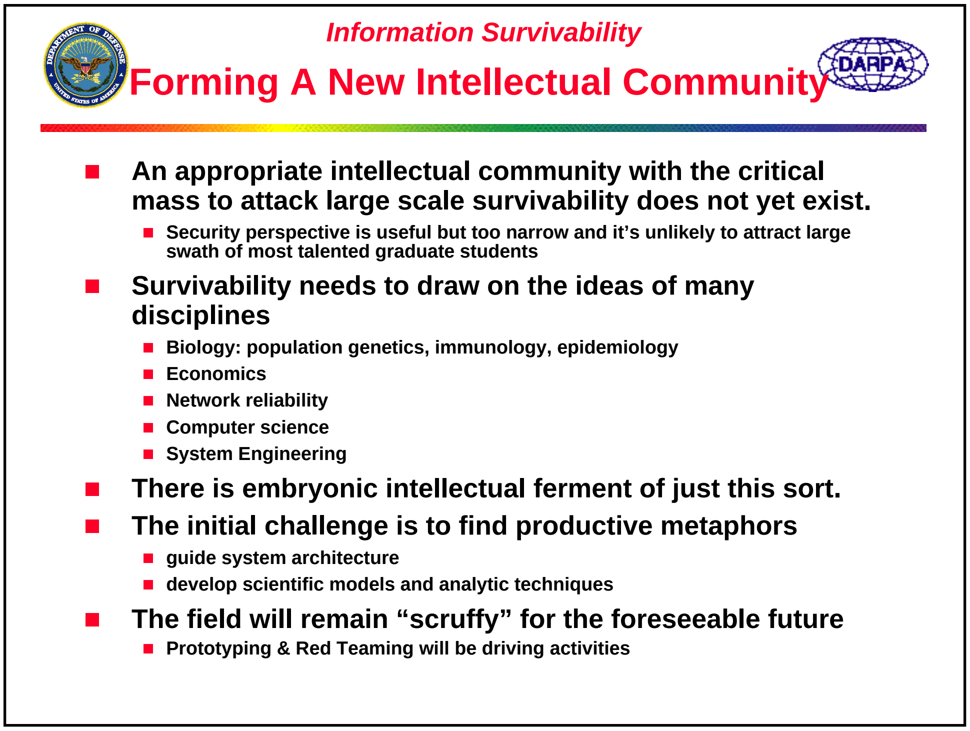
Diversity & Variability

- In biological ecologies, Monocultures are fragile
  - If all of Iowa is planted in one variety of corn, the right corn borer will do in the whole state. That variety of borer might not be known yet.
- Diversity reduces overall losses
  - At least some elements will survive and provide basis for reconstitution
  - Population evolves towards more survivable elements
- Variability hedges against unknown means of attack
- Economic forces make our computational ecology a monoculture
  - The Morris worm was successful because everything on the Net was a UNIX box (except my Lisp Machine!)
  - Vulnerabilities are often in the implementation not the design
- Challenge is to create a Polyculture within this uniformity

Finally, we should observe that in nature “monocultures” are the least survivable. If you plant a large area with the same variety of crop, then sooner or later a parasite will show up that will kill all of it. Polycultures, in which many varieties are intermixed will lose some individuals to man different forms of infection, but almost all individuals will survive almost all attacks. Statistically, a polyculture is never in very bad shape and the surviving elements can then reconstitute a whole population.

Unfortunately, our computer environments are much more like monocultures than is good for us. 90% of our desktop systems run the same (or very similar) operating systems; 90% of our servers run the same (or very similar) OS’s. There are perfectly good economic reasons for this so it’s not likely to change quickly; and if it does, we’ll probably still wind up with 90% of the systems running the same OS (it will just be a different one than the one they run today). Such uniformity is dangerous; if a good means of attack is discovered (and there are many today) then virtually all of our machines can be disabled quickly (remember how fast the Morris worm sped through the internet).

What we need are techniques that will engender variability among our computer systems even though there are strong economic forces towards uniformity.



Information Survivability

Forming A New Intellectual Community

- An appropriate intellectual community with the critical mass to attack large scale survivability does not yet exist.
  - Security perspective is useful but too narrow and it's unlikely to attract large swath of most talented graduate students
- Survivability needs to draw on the ideas of many disciplines
  - Biology: population genetics, immunology, epidemiology
  - Economics
  - Network reliability
  - Computer science
  - System Engineering
- There is embryonic intellectual ferment of just this sort.
- The initial challenge is to find productive metaphors
  - guide system architecture
  - develop scientific models and analytic techniques
- The field will remain "scruffy" for the foreseeable future
  - Prototyping & Red Teaming will be driving activities

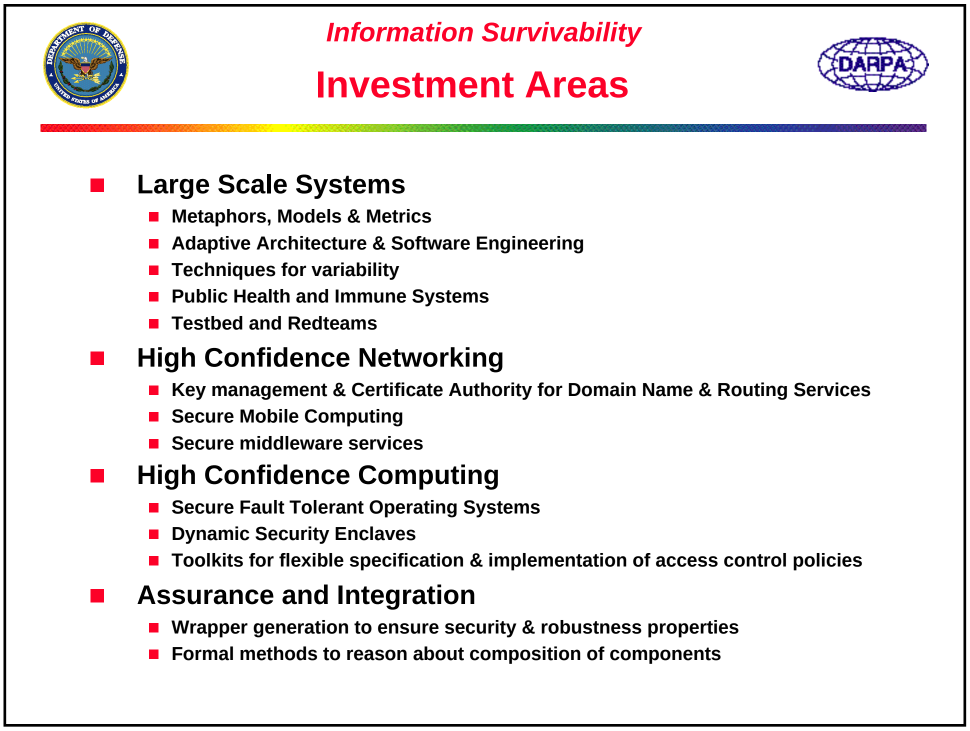
The ideas which we've presented are exciting, but they aren't today the core research ideas of any identifiable research community. We need to create that research community if we are to be successful and we invite you to join in what is likely to be a great intellectual adventure.

Survivability is about engineering a large scale, adaptive, self healing system. Many existing intellectual disciplines have useful things to say about this: among these are economics, population genetics, biological sciences in general, network reliability as studied by telephony and power system engineers and certainly many others.

In recent years, there has been a growing ferment and interest in a variety of areas that seem related to the synthesis we seek: the study of complex systems, artificial life, and emergent properties of collections of simple systems are some of these budding areas of interest.

To be successful we will need to find useful metaphors, to see how these metaphors lead to engineering methodologies, to analytic techniques and to useful metrics that help us understand if we are making progress.

But, as is often the case at the beginning of great intellectual adventures, it will take time for these ideas to settle down into a new paradigm with well understood methods.

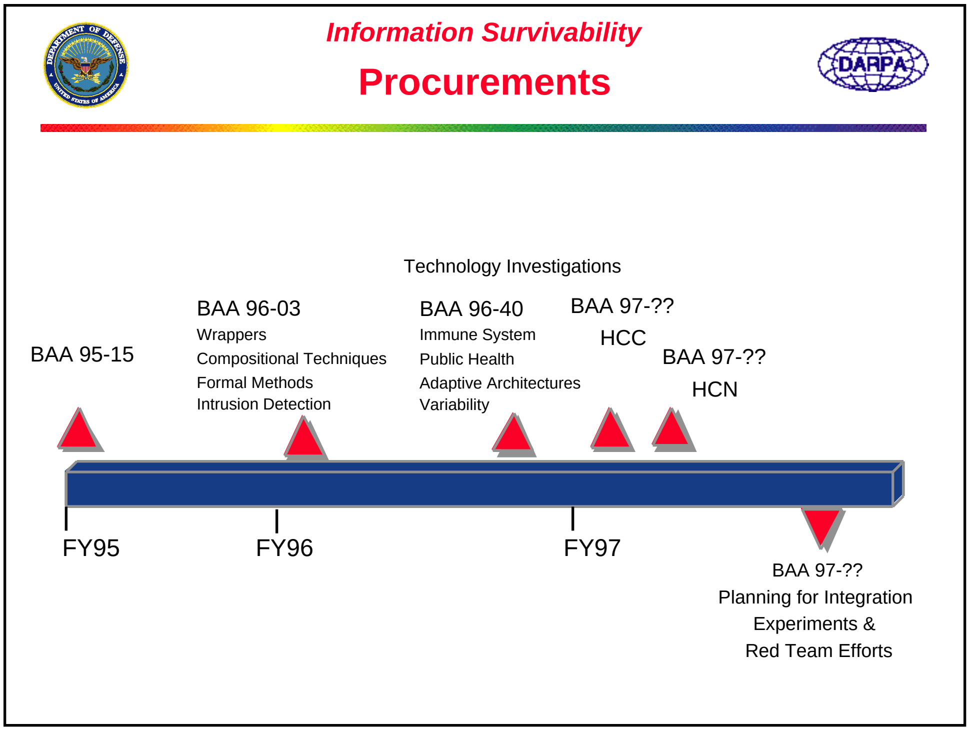


Information Survivability

Investment Areas

- Large Scale Systems
  - Metaphors, Models & Metrics
  - Adaptive Architecture & Software Engineering
  - Techniques for variability
  - Public Health and Immune Systems
  - Testbed and Redteams
- High Confidence Networking
  - Key management & Certificate Authority for Domain Name & Routing Services
  - Secure Mobile Computing
  - Secure middleware services
- High Confidence Computing
  - Secure Fault Tolerant Operating Systems
  - Dynamic Security Enclaves
  - Toolkits for flexible specification & implementation of access control policies
- Assurance and Integration
  - Wrapper generation to ensure security & robustness properties
  - Formal methods to reason about composition of components

As indicated earlier, the investments in the program are in four major areas. Here is a list of the technologies of interest in each of these areas.



Information Survivability

Procurements

Technology Investigations

BAA 95-15

BAA 96-03
Wrappers
Compositional Techniques
Formal Methods
Intrusion Detection

BAA 96-40
Immune System
Public Health
Adaptive Architectures
Variability

BAA 97-??
HCC

BAA 97-??
HCN

FY95

FY96

FY97

BAA 97-??
Planning for Integration
Experiments &
Red Team Efforts

This slide indicates the procurements that are currently underway or anticipated.